NASA'S MISSION TO PLANET EARTH

EARTH PROBES

DATA INFORMATION SYSTEM

EOS

EARTH OBSERVING SYSTEM

FOS Prototype Results Review

Andy Miller

24 August 1995

Prototype Results Review
Agenda

Prototype Results Overview	Andy Miller
Prototype Results Presenters	
• User Interface	Jim Creegan
• Planning and Scheduling	Guy Swope
- Resource Model Distribution	Guy Swope
- Accesses	Ken Cockerill
- Activity Definition	Jim Yu
- Activity Constraints	Steve Pawlish
• Command Management	Theresa Brandt
• Real-time	Ken Fregeolle
• Analysis	Kevin Robair
• Data Management	Dave Peters

FOS Prototype and Studies Background

General FOS prototype goals

- User satisfaction
  - Demos to the user community (i.e., formal, informal)
  - Useability testing
  - Involvement of the ECS FOT (e.g., SSR rule definition)
- Risk mitigation
  - Drive out requirements and design
  - Develop object oriented methodology experience
  - Development of interfaces (e.g., Planning and Scheduling and Command Management)
- Technology Assessment
  - Expert Systems
  - User interface features (Schematics, 3D plots)

FOS Prototype and Studies Background

FOS prototyping initially performed as four individual prototypes

- Planning and Scheduling
- Command Management
- Decision Support
- Instrument Support Toolkit

FOS conducted Prototype Results Reviews demonstrating the FOS prototypes and delineating results

- PRR #1 November 1993
- PRR #2 June 1994
- PRR #3 February 1995

FOS Prototype and Studies Goals

Primary Theme for February 1995 through August 1995 phase

- Perform studies and analysis to drive out the FOS detailed design
- Solve specific design issues through proof-of-concept prototypes

Key Objectives

- Determine COTS to procure
  - User interface graphics
  - Analysis packages
  - Data Base package
- Analyze detailed design issues
- Prepare and refine screen mock-ups

System Prototypes

ECS class library selection

Purpose

- Evaluate and select the class library for the ECS project

Approach

- Evaluate class libraries (Booch Components, Hughes Class Libraries, Object Systems, Rogue Wave)
- Evaluation criteria
  - POSIX compliant, thread safe, exception handling built-in, multi-platform support, add-on libraries, ease of use with IPC, multiple inheritance, templates
- Evaluation process
  - 8 developers across ECS project; 2 per library
  - Same development/maintenance task for every library
  - Evaluation data scored (time to find a class, time to use a class, documentation, time to modify an existing application)

System Prototypes

Results

- Selected Rogue Wave
- Clear choice using both objective and subjective criteria

System Prototypes

Prioritize coding

- General Results
  - Identify and develop common software objects first
  - Subclasses are inherited from fully defined base classes

Stp/OMT Code Generation Capabilities

- General Results
  - Generates templates from object models from which software development can start
  - Reduces menial typing
  - Automate data dictionary generation from OMT-generated header files

System Prototypes

Low-level software items

- General Results
  - Prototyped sending Rogue Wave objects across sockets, DECmsg Queues, Pure Logic Pipes

    Mitigate risk of passing classes

    Verified ability to pass classes in each case
  - Prototyped multiple inheritance with Rogue Wave

    Determine if multiple inheritance supported on all platforms

    Multiple inheritance is fully supported; increases reuse

System Prototypes - On-going and Future

NSI Testbed/IST simulation prototype

Purpose

- **Determine/analyze EOC/IST network performance characteristics**
- **Scheduling**
  - **Interactive scheduling**
- **Real-time**
  - **Packet sequence**
  - **Packet delay**
  - **Data transfer rates (e.g., telemetry)**
- **File transfers**
  - **FTP**

System Prototypes - On-going and Future

Parameter server/monitor prototype

Purpose

- Provide generic mechanism for the maintenance, service, and receipt of ground system and telemetry parameters
- Used by multiple FOS subsystems (i.e., user interface, telemetry, command, and analysis)

PC-based Unix User Station Prototype

Purpose

- Evaluate potential porting and performance issues regarding low-cost workstation solutions (i.e., Solaris)